

1/

```
CREATE OR REPLACE PROCEDURE
display_project(v_nom employe.nom%TYPE, v_prenom employe.prenom%TYPE)
IS
CURSOR joint IS SELECT * FROM employe
WHERE nom=v_nom AND prenom=v_prenom
INNER JOIN travaille WHERE matricule ON employe.matricule=travaille.matricule;
jointure joint%rowtype;
BEGIN
OPEN joint;
LOOP
    FETCH joint INTO jointure;
    EXIT WHEN joint%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('Prenom : '||jointure.prenom||'Nom : '
||jointure.nom||'Projet : '||jointure.NomProjet||);
END LOOP;
CLOSE joint;
END;
```

2/

```
CREATE OR REPLACE function nombre_projets(matricule_type employe.matricule%TYPE)
RETURN NUMBER IS
nombreProjet NUMBER;
BEGIN
    SELECT count(NomProjet)
    INTO nombreProjet FROM travaille
    WHERE matricule=matricule_type;
    RETURN nombreProjet;
END;
```

3/

```
CREATE OR REPLACE function nombre_heures(matricule_type_bis
employe.matricule%TYPE)
RETURN NUMBER IS nb_heures NUMBER;
BEGIN
    SELECT sum(nbHeure)
    INTO nb_heures FROM travaille
    WHERE matricule = matricule_type_bis;
    RETURN nb_heures;
END;
```

4/

```
TYPE employe IS RECORD(
    nom VARCHAR2(30),
    prenom VARCHAR2(50)
);
TYPE tab_employe IS TABLE OF employe
INDEX BY BINARY_INTEGER;
tab tab_employe;
CURSOR project(the_projet projet.nomProjet%TYPE) IS SELECT nom,prenom FROM
employe
WHERE matricule = ANY(SELECT matricule FROM travaille WHERE nomProjet =
the_projet);
project monProjet%rowtype;
leProjet projet.nomProjet%TYPE;
j NUMBER;
i NUMBER;
unknownProject EXCEPTION;
BEGIN
leprojet:='&nomProjet';
OPEN project;
i:=1;
LOOP
    FETCH project INTO monProjet;
    EXIT WHEN project%notfound;
    tab(i).nom = monProjet.nom;
    tab(i).prenom = monProjet.prenom;
    i := i + 1;
END LOOP;
For j in tab.FIRST..tab.Last
LOOP
    Dbms_output.put_line ("Nom : "||tab(j).nom||" Prenom : "||tab(j).prenom);
END LOOP;
CLOSE project;
EXCEPTION
WHEN unknownProject THEN
dbms_output.put_line('Erreur! Projet inexistant...');
END;/
```

5/

```
BEGIN
    SELECT * FROM projet
    WHERE nomProjet not in(SELECT nomProjet FROM travaille);
END;/
```

6/

```
BEGIN
    SELECT nomProjet FROM travaille
    GROUP BY nomProjet
    HAVING count(nomProjet)=ANY(SELECT count(matricule) FROM employe);
END;
```

{ Construction des tables }

```
CREATE TABLE travaille(  
    matricule INT NOT NULL,  
    NomProjet varchar(15) NOT NULL,  
    nbHeure INT NOT NULL,  
    CONSTRAINT fk_matricule FOREIGN KEY (matricule) REFERENCES  
employe(matricule)ON DELETE CASCADE,  
    CONSTRAINT fk_nom FOREIGN KEY (NomProjet) REFERENCES projet(NomProjet)ON  
DELETE CASCADE,  
    CONSTRAINT pk_travaille PRIMARY KEY (matricule,NomProjet)  
);  
  
INSERT INTO employe(nom, prenom)  
VALUES("dzadou","Dravidio"),  
("davis","ben"),  
("kante","mamadouss"),  
("Aminata","Diop");  
  
INSERT INTO projet(NomProjet, description)  
VALUES("PDO", "A PHP simple project"),  
("PlocX", "An unknown being outta here"),  
("Hitman", "An assassin game"),  
("Tiakola", "A MHD song");  
  
INSERT INTO travaille(matricule, NomProjet, NbHeure)  
VALUES(10023, "PDO", 6),  
(1102023, "PlocX", 8),  
(100232, "Hitman", 4),  
(310023, "Tiakola", 3),  
(100231, "Laptop", 2),  
(210023, "Dammit", 3),  
(100232, "Hore", 4),  
(210023, "Hoe", 4);
```